Phase 3: Development Part 1

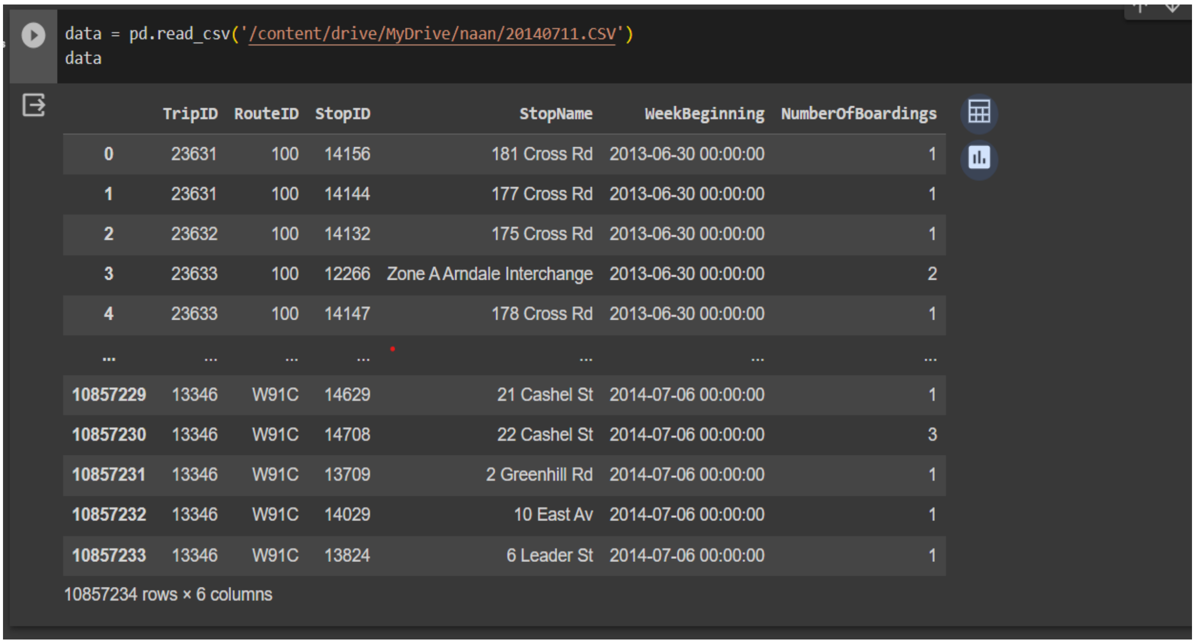
Problem: Public Transport Efficiency Analysis

Preprocessing

Preprocessing is the essential initial phase in data analysis and machine learning. It involves cleaning and organizing raw data to ready it for analysis or model training. This includes handling missing values, transforming data for consistency, reducing dimensionality, and converting text or image data into suitable formats. Preprocessing tackles issues like outliers, imbalances, and noise. Effective preprocessing ensures accurate results and efficient utilization of machine learning algorithms, making data more accessible and informative for subsequent analytical processes. It enhances the quality and reliability of insights and predictions derived from the data.

Preprocessing is carried out in the given data set using python library pandas. The following preprocessing steps has been carried out in the dataset :

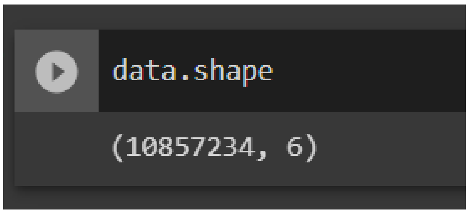
1. Loading the dataset from the csv file using read_csv method of pandas.



```
data = pd.read_csv('/content/drive/MyDrive/naan/20140711.CSV')
data
```

```
          TripID  RouteID  StopID                StopName         WeekBeginning  NumberOfBoardings
       0   23631      100   14156            181 Cross Rd   2013-06-30 00:00:00                  1
       1   23631      100   14144            177 Cross Rd   2013-06-30 00:00:00                  1
       2   23632      100   14132            175 Cross Rd   2013-06-30 00:00:00                  1
       3   23633      100   12266  Zone A Arndale Interchange  2013-06-30 00:00:00               2
       4   23633      100   14147            178 Cross Rd   2013-06-30 00:00:00                  1
     ...     ...      ...     ...                     ...                   ...                ...
10857229   13346     W91C   14629            21 Cashel St   2014-07-06 00:00:00                  1
10857230   13346     W91C   14708            22 Cashel St   2014-07-06 00:00:00                  3
10857231   13346     W91C   13709           2 Greenhill Rd  2014-07-06 00:00:00                  1
10857232   13346     W91C   14029              10 East Av   2014-07-06 00:00:00                  1
10857233   13346     W91C   13824             6 Leader St   2014-07-06 00:00:00                  1
10857234 rows × 6 columns
```

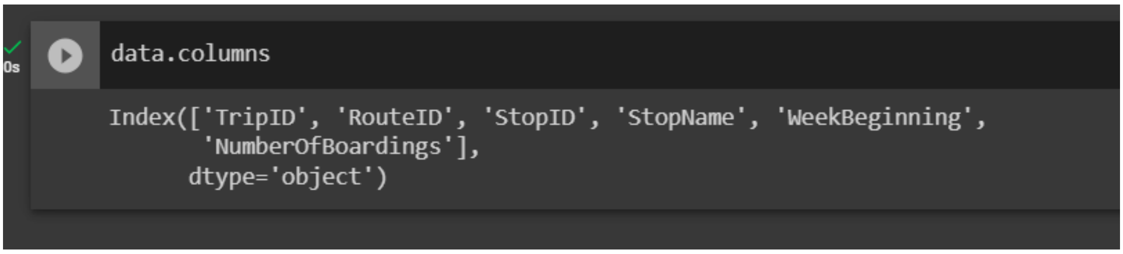
2. Viewing the shape of the given dataset



```
data.shape
```

```
(10857234, 6)
```

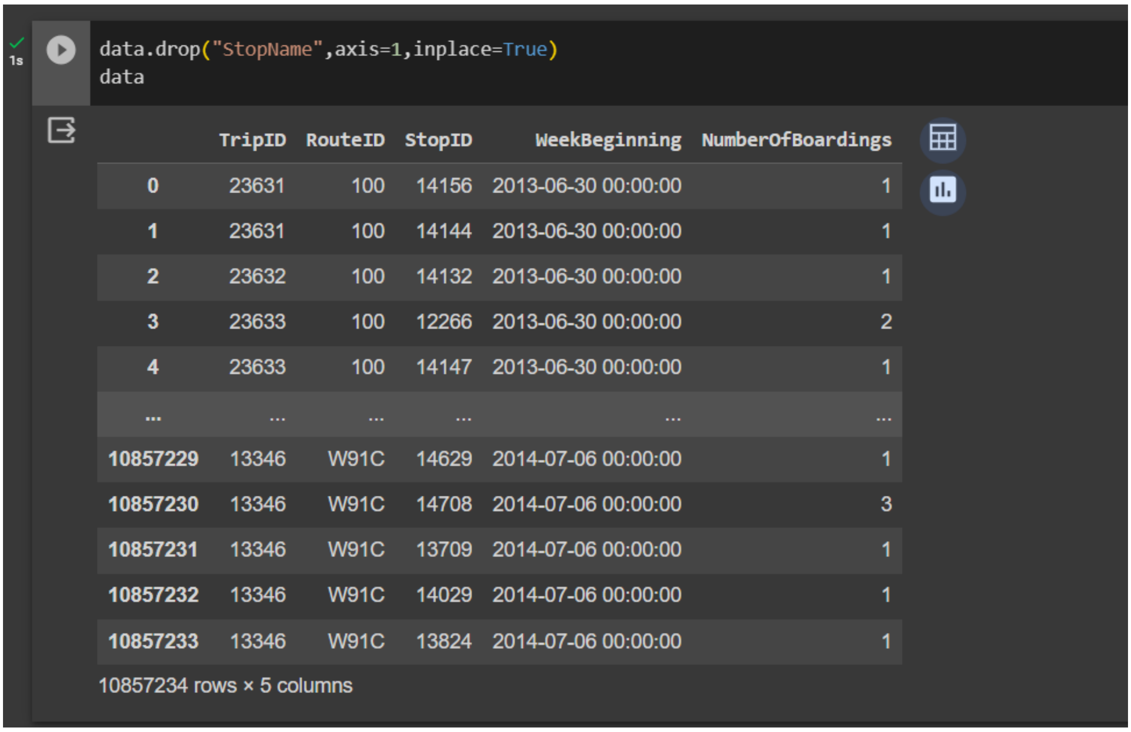
3. Viewing the columns of the dataset



```
data.columns
```

```
Index(['TripID', 'RouteID', 'StopID', 'StopName', 'WeekBeginning',
       'NumberOfBoardings'],
      dtype='object')
```

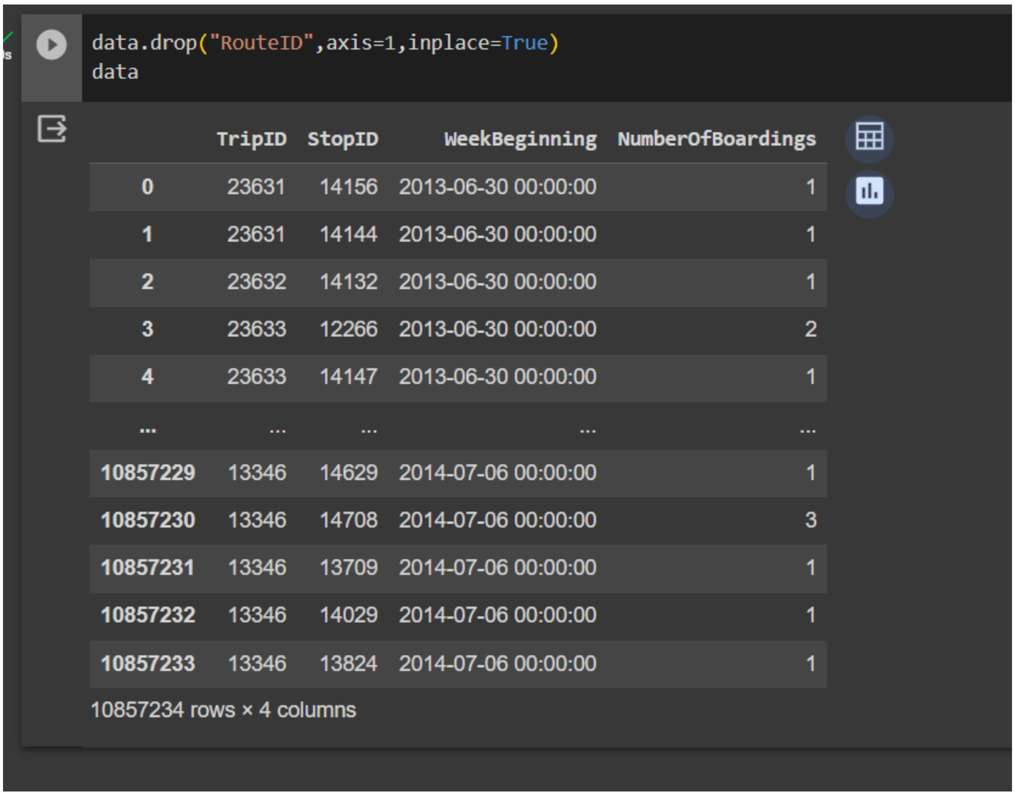
4. Dropping the ***StopName*** column as it is not needed for analysis using drop method.



```
data.drop("StopName",axis=1,inplace=True)
data
```

```
          TripID  RouteID  StopID        WeekBeginning  NumberOfBoardings
0          23631      100   14156  2013-06-30 00:00:00                  1
1          23631      100   14144  2013-06-30 00:00:00                  1
2          23632      100   14132  2013-06-30 00:00:00                  1
3          23633      100   12266  2013-06-30 00:00:00                  2
4          23633      100   14147  2013-06-30 00:00:00                  1
...          ...      ...     ...                  ...                ...
10857229   13346     W91C   14629  2014-07-06 00:00:00                  1
10857230   13346     W91C   14708  2014-07-06 00:00:00                  3
10857231   13346     W91C   13709  2014-07-06 00:00:00                  1
10857232   13346     W91C   14029  2014-07-06 00:00:00                  1
10857233   13346     W91C   13824  2014-07-06 00:00:00                  1
10857234 rows × 5 columns
```

5. Dropping ***RouteID*** as it is not needed for analysis

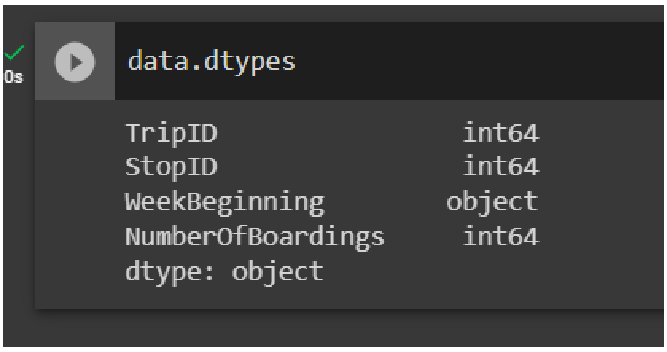


```
data.drop("RouteID",axis=1,inplace=True)
data
```

	TripID	StopID	WeekBeginning	NumberOfBoardings
0	23631	14156	2013-06-30 00:00:00	1
1	23631	14144	2013-06-30 00:00:00	1
2	23632	14132	2013-06-30 00:00:00	1
3	23633	12266	2013-06-30 00:00:00	2
4	23633	14147	2013-06-30 00:00:00	1
...	...	...	...	...
10857229	13346	14629	2014-07-06 00:00:00	1
10857230	13346	14708	2014-07-06 00:00:00	3
10857231	13346	13709	2014-07-06 00:00:00	1
10857232	13346	14029	2014-07-06 00:00:00	1
10857233	13346	13824	2014-07-06 00:00:00	1

10857234 rows × 4 columns

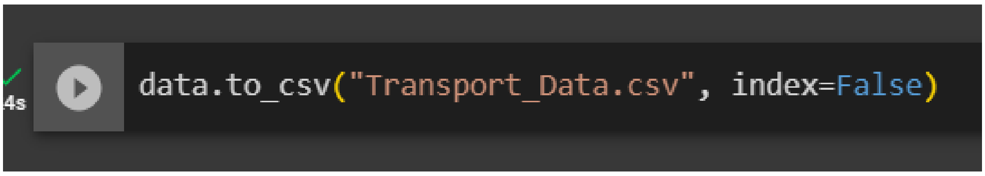
6. Viewing the datatypes after preprocessing



```
data.dtypes
```

```
TripID                int64
StopID                int64
WeekBeginning        object
NumberOfBoardings     int64
dtype: object
```

7. Storing the preprocessed data to a csv file.

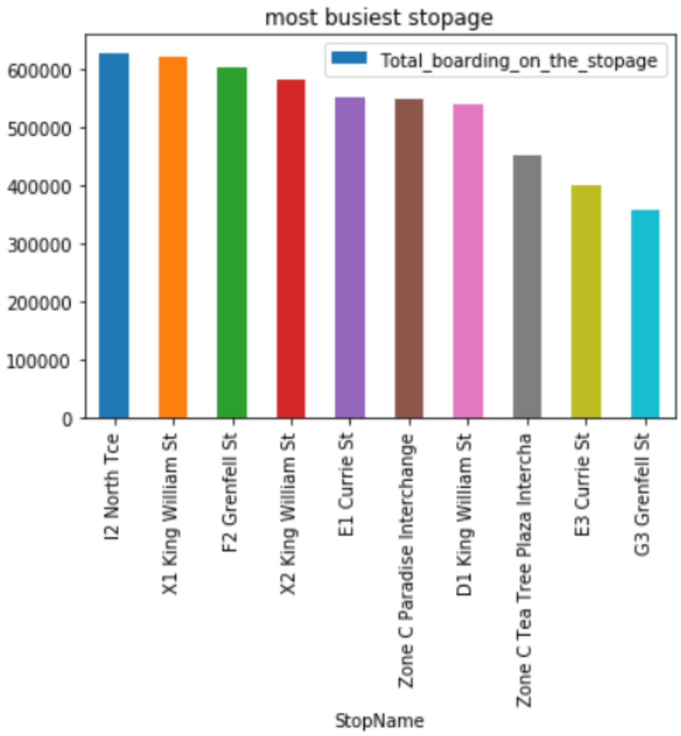


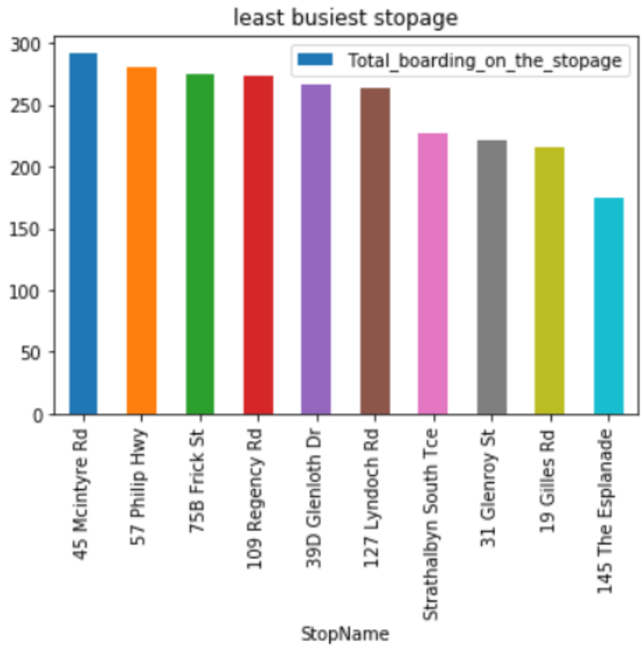
```
data.to_csv("Transport_Data.csv", index=False)
```

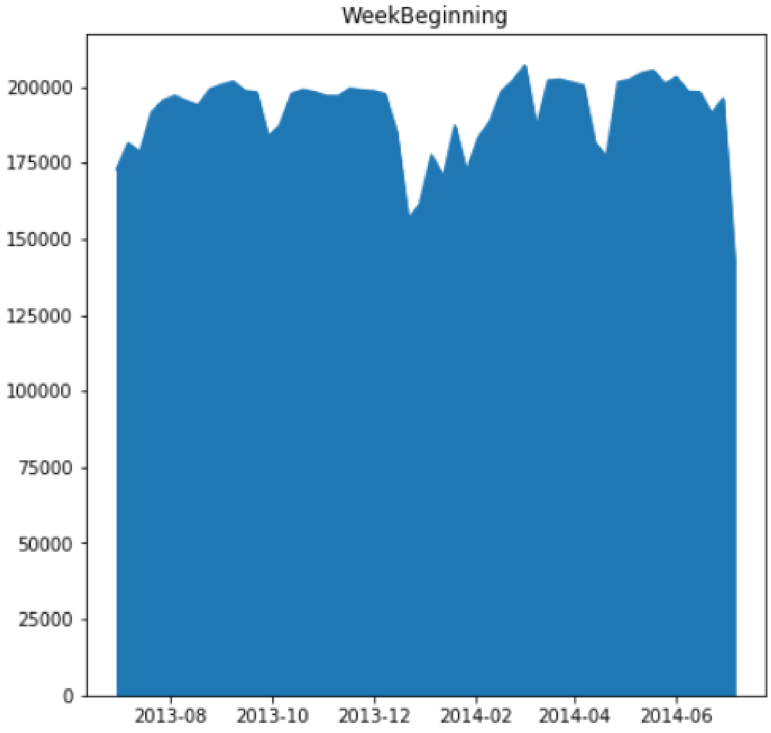
Data Visualization

Data visualization is a powerful technique to represent data in a graphical or visual format, making complex information easily understandable. It involves creating visual representations like charts, graphs, and maps to uncover patterns, trends, and insights within data. Through color, shape, and layout, data visualization helps convey information rapidly and efficiently, aiding decision-making and storytelling. Effective data visualization enhances communication, enabling stakeholders to grasp complex data relationships, outliers, and correlations. It is a vital tool for analysts, researchers, and decision-makers to present findings, explore data, and derive meaningful conclusions for informed actions and strategies.

The dataset after preprocessing is loaded to IBM cognos to generate the visualization.







Conclusion:

Thus the given dataset was preprocessed and relevant graphs were plotted using IBM cognos.